

**IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
AUSTIN DIVISION**

ANCORA TECHNOLOGIES, INC.,

Plaintiff,

v.

LG ELECTRONICS INC. and LG
ELECTRONICS U.S.A., INC.,

Defendants.

CIVIL ACTION NO. 1:20-cv-0034

JURY TRIAL DEMANDED

ANCORA TECHNOLOGIES, INC.,

Plaintiff,

v.

SAMSUNG ELECTRONICS CO., LTD. and
SAMSUNG ELECTRONICS AMERICA,
INC.,

Defendants.

CIVIL ACTION NO. 1:20-cv-0034

JURY TRIAL DEMANDED

DEFENDANTS' OPENING CLAIM CONSTRUCTION BRIEF

Table of Contents

I.	INTRODUCTION	1
II.	LITIGATION HISTORY	4
III.	ARGUMENT	5
A.	“using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS”	5
1.	The term “agent” is a “nonce” word.	6
2.	The specification and the prosecution history confirm that “agent” does not connote structure.	8
3.	The associated algorithm is disclosed at 6:18-28 in the patent.....	10
B.	“set up a verification structure”	11
C.	“memory of the BIOS”	14
1.	The applicants disclaimed a “memory of the BIOS” that is recognized by an operating system as a storage device or that contains a file system.	15
2.	The “memory of the BIOS” is the memory that stores the BIOS.	18
D.	“verifying the program using at least the verification structure”	18
E.	“acting on the program according to the verification”	20
F.	“license”/“license record”	21
1.	The preamble of the claim is limiting.	21
2.	License	23
3.	License Record.....	24
G.	Order of Steps	24
H.	“BIOS”	26
1.	“BIOS” is stored in ROM.	26
2.	“BIOS” runs automatically when a computer is powered on.	27
I.	“selecting a program residing in the volatile memory”	28

J.	“program”	28
K.	“volatile memory”.....	29
L.	“first non-volatile memory area of the computer”	30

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>Altiris, Inc. v. Symantec Corp.</i> , 318 F.3d 1363 (Fed. Cir. 2003).....	25
<i>Ancora Techs., Inc. v. Apple, Inc.</i> , 744 F.3d 732 (Fed. Cir. 2014).....	<i>passim</i>
<i>Arendi S.A.R.L. v. Google LLC</i> , 882 F.3d 1132 (Fed. Cir. 2018).....	18
<i>Bell Atlantic Network Servs., Inc. v. Covad Commc'ns Group, Inc.</i> , 262 F.3d 1258 (Fed. Cir. 2001).....	14
<i>Biedermann Motech GmbH v. Acme Spine, LLC</i> , No. CV 06-3619 SJO, 2007 WL 6210841 (C.D. Cal. Aug. 31, 2007)	26
<i>C.R. Bard, Inc. v. U.S. Surgical Corp.</i> , 388 F.3d 858 (Fed. Cir. 2004).....	11
<i>C.W. Zumbiel Co. v. Kappos</i> , 702 F.3d 1371 (Fed. Cir. 2012).....	22
<i>Cypress Lake Software, Inc. v. Samsung Elecs. Am., Inc.</i> , 382 F. Supp. 3d 586, 614-17 (E.D. Tex. 2019).....	8
<i>Digital Retail Apps Inc. v. H-E-B, LP</i> , No. 6-19-cv-00167-ADA, 2020 WL 376664 (W.D. Tex. Jan. 23, 2020).....	8, 10
<i>Fenner Investments, Ltd. v. Cellco P'ship</i> , 778 F.3d 1320 (Fed. Cir. 2015).....	14
<i>Genband USA v. Metaswitch Networks</i> , No. 2:14-cv-33-RG-RSP, 2015 WL 4722185 (E.D. Tex. Aug. 7, 2015)	8
<i>Global Equity Mgmt. (SA) Pty. Ltd. v. Expedia, Inc.</i> , No. 2:16-cv-00095-RWSRSP, 2016 WL 7416132 (E.D. Tex. Dec. 22, 2016)	8
<i>Honeywell Int'l, Inc. v. ITT Indus., Inc.</i> , 452 F.3d 1312 (Fed. Cir. 2006).....	14, 20
<i>Joao Control & Monitoring Systems LLC v. Protect America, Inc.</i> , No. 1-14-cv-134, 2015 WL 4937464, at *7 (W.D. Tex. Aug. 18, 2015)	7, 8

<i>Kaken Pharm. Co. v. Iancu</i> , No. 2018-2232, 2020 WL 1222728 (Fed. Cir. Mar. 13, 2020).....	15, 18
<i>Mantech Envtl. Corp. v. Hudson Envtl. Servs., Inc.</i> , 152 F.3d 1368 (Fed. Cir. 1998).....	25
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005).....	12, 23
<i>Pitney Bowes, Inc. v. Hewlett-Packard Co.</i> , 182 F.3d 1298 (Fed. Cir. 1999).....	22
<i>Poly-Am., L.P. v. API Indus., Inc.</i> , 839 F.3d 1131 (Fed. Cir. 2016).....	12, 20
<i>Synchronoss Techs., Inc. v. Dropbox Inc.</i> , No. 16-cv-00119, 2017 WL 6059302 (N.D. Cal. Dec. 7, 2017).....	7
<i>Techno View IP, Inc. v. Facebook Techs., LLC</i> , No. 17-386-CFC-CJB, 2018 WL 6427874 (D. Del. Dec. 7, 2018)	10
<i>Verint Sys. Inc. v. Red Box Recorders Ltd.</i> , 166 F. Supp. 3d 364, 379-381 (S.D.N.Y. 2016)	8
<i>ViaSat, Inc. v. Space Sys./Loral, Inc.</i> , No. 3:12-CV-00260-H, 2013 WL 3927729 (S.D. Cal. May 29, 2013)	18
<i>Williamson v. Citrix Online</i> , 792 F.3d 1339, 1349 (Fed. Cir. 2015).....	6, 10
Statutes	
35 U.S.C. § 112 ¶ 6.....	5, 6, 7, 8

I. INTRODUCTION

The '941 Patent, entitled “Method of Restricting Software Operation within a License Limitation,” describes a method of identifying and restricting unauthorized use of a software program on a computer. Ex. 1 ('941 Patent) at 1:6-8. The patent is directed to preventing software piracy by verifying that software programs are licensed. *Id.* This is accomplished by creating a distinct identifier for a particular computer that can be used to verify whether a program is licensed for use on that computer. *See, e.g., id.* at 2:27-59.

The “Summary of the Invention” explains how the invention operates. Initially, a verification structure is set up in the BIOS memory of the computer to indicate that a specified program is licensed to run on the computer.¹ *Id.* at 1:59-62. The verification structure is formed by encrypting a license record associated with the program using a key that uniquely identifies the computer. The key is stored in the BIOS memory. *Id.* at 1:62-67. Later when a user wants to use that program, a license verifier application verifies whether the program is licensed to run on the computer. *Id.* at 2:10-26. The license verifier accesses the program, retrieves the license record from the program, and encrypts the license record using the key of the computer. *Id.* It then compares the encrypted license record with the encrypted records previously stored in the BIOS memory. *Id.* In case of a match, the program is found to be licensed and is allowed to run on the computer. Otherwise, appropriate action, including potential restriction, is taken. *Id.*

The patent acknowledges that software-based and hardware-based methods to determine whether a program on a computer is licensed were already known in the art, but explains that these methods were purportedly deficient. *Id.* at 1:19-32. The patent claims that its method overcomes

¹ Figure 1 of the '941 Patent depicts a computer (1) that contains a non-volatile memory area of the BIOS (4 & 5) and a volatile memory area (6) (i.e., the internal RAM memory). The BIOS memory has two portions, a first non-volatile memory area (4) (e.g., the ROM section) and a second non-volatile memory area (5) (e.g., the E²PROM section).

these deficiencies based on two specific improvements. *First*, the invention uses a key that is unique to the computer to encrypt the verification structure, allowing any attempt to run the program on an unlicensed computer to be detected immediately. *Id.* at 2:27-59. *Second*, the invention stores the encrypted verification structure in the BIOS memory—a memory space that the patent describes as being more difficult to access and tamper with than storage areas used by earlier methods. *Id.* 1:46-2:5; 3:4-14; 4:45-48.

During prosecution, the applicants made extensive arguments based on these two alleged improvements to differentiate the alleged invention from the prior art and to gain allowance. Mirroring the specification, the applicants made clear that several concepts described below were fundamental to the invention.

Encrypting the verification structure using a pseudo-unique key: The Examiner initially rejected the pending claims as anticipated by US 5,892,900 (“Ginter”). The applicants argued that Ginter did “not anticipate the present invention” because it “d[id] not disclose . . . setting up a verification structure.” Ex. 4 (Response, May 23, 2001, p. 6). In particular, the applicants highlighted that the “present invention” overcomes deficiencies associated with Ginter because “[i]n the present method, the verification structure is formed using a unique key for each computer.” *Id.* at p. 8. The applicants subsequently confirmed that the “pseudo-unique key recited in claims 1 and 20” is crucial to the invention. Ex. 8 (Response, Feb. 5, 2002, p. 3) (emphasis in original).

BIOS memory is not recognized by the operating system and does not have a file system: The Examiner also rejected the pending claims based on US 6,189,146 (“Misra”) in view of US 5,479,639 (“Ewertz”). To overcome the rejection, the applicants argued that by using BIOS memory to store application data, “the ***present invention*** proceeds against conventional wisdom

in the art.” Ex. 8 (Response, Feb. 5, 2002, p. 6) (emphasis added). The applicants argued that an “ordinary skilled artisan would not consider the BIOS as a storage medium to preserve application data for at least two reasons.” *Id. First*, “[t]here is no OS support whatsoever to write data to the system BIOS,” and, *second*, “no file system is associated with the BIOS.” *Id.* These features were “further evidence that OS level application programmers would not consider the BIOS as a storage medium for license data.” *Id.* Echoing the specification, the applicants underscored the benefits of using the claimed BIOS memory that is generally inaccessible, as opposed to more readily accessible memory.

The invention operates at the OS-level: In arguing against the Misra and Ewertz combination, the applicants further characterized their invention as operating at the OS-level as opposed to the BIOS-level. The applicants argued that “[s]oftware license management applications, such as the one of the ***present invention***, are operating system (OS) level programs.” Ex. 8 (Response, Feb. 5, 2002, p. 5) (emphasis added). Critically, the Examiner relied upon the applicants’ characterization in the Notice of Allowance. Ex. 9 (Notice of Allowance dated Mar. 28, 2002, Reasons for Allowance, p. 4).

Using an agent: The applicants acknowledged during prosecution that BIOS memory was well-known in the art, as were OS-level methods of verifying that a program was licensed. The applicants argued that the OS-level and the BIOS-level are “mutually exclusive” and “cannot run at the same time.” Ex. 8 (Response, Feb. 5, 2002, p. 5). The purported breakthrough in applicants’ invention was to somehow allow the OS-level programs to write to the BIOS memory. Although the specification nowhere refers to or describes an “agent,” the applicants added “agent” to the claims and portrayed agent as a novel concept, hitherto unknown in prior art, to fill this void. When allowing the claims, the Examiner pointed to the problem in prior art that “a computer BIOS is not

setup to manage a software license verification structure” but noted that the “*present invention* overcomes this difficulty by using an agent to set up a verification structure in the . . . memory of the BIOS.” Ex. 9 (Notice of Allowance dated Mar. 28, 2002, Reasons for Allowance, p. 4) (emphasis added).

Ancora now takes a shifting sands approach, walking away from the representations the applicants made to the USPTO, as well as the positions it has taken in other litigations, proposing that the terms should simply be afforded their “plain and ordinary meaning.” Ancora ignores the fact that applicants told the USPTO that the invention uses a pseudo-unique key to encrypt the verification structure, ignores the fact that the applicants told the USPTO that BIOS memory is not recognized by an operating system and does not have a file system, ignores the fact that the applicants told the USPTO that the invention operates at the OS-level, and ignores the fact that the USPTO relied on those arguments as the basis for allowance. Ancora even walks away from the alleged heart of the invention, the “agent,” which Ancora now contends can simply be any “software program or routine.” That approach will not aid the jury. Nor should Ancora be allowed to renege on the bargain the applicants made with the USPTO to receive their patent.

II. LITIGATION HISTORY

The ’941 Patent was first asserted in 2008 against Toshiba America Information Systems, et al., and Microsoft Corporation successfully intervened in that case. *See* C.A. No. 8:08-cv-00626 (Central District of California) (the “Microsoft Litigation”) at ECF Nos. 1, 49. Just before dismissal, the parties filed their respective claim construction briefs. *Id.* at ECF Nos. 101, 102.

Ancora next filed suit against Apple in 2010. *See* C.A. No. 2:10-cv-10045 (Central District of California) at ECF No. 1. The case was transferred to the Northern District of California (C.A. No. 4:11-cv-06357) (the “Apple Litigation”) where the parties engaged in claim construction proceedings, resulting in a Markman order. *See* Apple Litigation at ECF Nos. 94,

96, 107. A number of the constructions were ultimately appealed to the Federal Circuit. *See Ancora Techs., Inc. v. Apple, Inc.*, 744 F.3d 732, 736 (Fed. Cir. 2014).

Ancora recently filed suit against HTC America, Inc. and HTC Corporation in the Western District of Washington (the “HTC Litigation”) on December 15, 2016. *See* C.A. No. 2:16-cv-01919 at ECF No. 1. The parties have fully briefed their respective claim construction positions (*see* HTC Litigation at ECF Nos. 59, 60, 62, 63), however no *Markman* order has been issued.

III. ARGUMENT

A. “using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS”

Defendants’ Proposal	Plaintiff’s Proposal
<p>This limitation is a means plus function limitation governed by pre-AIA 35 U.S.C. § 112 ¶ 6.</p> <p>Function: “set up a verification structure in the erasable, non-volatile memory of the BIOS”</p> <p>Structure: Algorithm found at 6:18-28; if not, indefinite due to a lack of corresponding structure</p>	<p>plain and ordinary meaning</p> <p>“agent” = “software program or routine”</p>

The applicants claimed that the heart of their alleged invention was writing an encrypted verification structure in the BIOS memory, and portrayed that the “agent,” a concept hitherto unknown in the prior art, would accomplish that. Based on the applicants’ portrayal, the Examiner noted that use of an “agent” was crucial to the alleged invention because the “agent” overcame the perceived difficulty of setting up a verification structure in the BIOS memory. Ex. 9 (Notice of Allowance dated Mar. 28, 2002, Reasons for Allowance, p. 4). Ancora’s proposal to give the term its “plain and ordinary meaning” and to construe “agent” merely as any “software program or routine” contradicts the applicants’ prior depiction of the term as a novel concept.

The term “agent” was added during prosecution to overcome a § 112 rejection that the

applicants had not disclosed the necessary hardware to “add, remove and modify a license record” in the BIOS memory. Ex. 5 (June 22, 2001, Final Rejection, pp. 3-6).² When amending claim 1, the applicants described the term “agent” in purely functional terms, i.e., the “agent” “set[s] up a verification structure in the erasable, non-volatile memory of the BIOS.” While pre-AIA 35 U.S.C. § 112 ¶ 6 allows an element in a claim to “be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof,” when an applicant chooses to claim in such a manner, the “claim shall be construed to cover the corresponding structure, materials, or acts described in the specification and equivalents thereof.” Here, the associated structure is recited in the ’941 Patent at 6:18-28, the only portion of the specification that describes an algorithm for setting up the verification structure of claim 1. Ancora’s proposal is an improper attempt to interpret the claims to cover any and all ways of performing the claimed function, in contravention of 35 U.S.C. § 112 ¶ 6.

1. The term “agent” is a “nonce” word.

The claims at issue do not use the terms “means” or “step for.” Hence, the Defendants must demonstrate that the claim term fails to “‘recite sufficiently definite structure’ or else recites ‘function without reciting sufficient structure for performing that function’” in order for 35 U.S.C. § 112 ¶ 6 to apply. *Williamson*, 792 F.3d at 1349. In *Williamson*, the Federal Circuit held that although the term “distributed learning control module” did not expressly invoke the word “means,” the term “[m]odule” is a well-known nonce word that can operate as a substitute for ‘means’ in the context of § 112, para. 6.” *Id.* at 1348-54. The court went on to explain that “‘module’ is simply a generic description for software or hardware that performs a specified

² Notably, the ’941 Patent was prosecuted before the Federal Circuit in *Williamson v. Citrix Online* “expressly overrule[d] the characterization of [the presumption that a limitation lacking the word ‘means’ is not subject to § 112] as strong.” 792 F.3d 1339, 1349 (Fed. Cir. 2015).

function,” and that “the word ‘module’ does not provide any indication of structure,” but rather “sets forth the same black box recitation of structure for providing the same specified function as if the term ‘means’ had been used.” *Id.* at 1350. Like “module,” the word “agent” is a nonce word that can be substituted for “means.” There is no substantive difference between a “*means for setting up a verification structure*” and “*using an agent to set up a verification structure.*”

Because the term “agent” is a nonce word, it does not provide any indication of structure to a person of ordinary skill in the art. Defendants’ expert, Dr. Erez Zadok, confirms that a person of ordinary skill in the art would not understand the term “agent” to connote sufficient structure. Zadok Decl. at ¶¶ 52-60. While the term “agent” can be found in dictionaries, the dictionaries provide disparate definitions for the term. As illustrated by the Dictionary of Computing, agents “may be software, hardware, or both.” *Id.* The term “agent” does not connote a particular structure that would set up a verification structure in the erasable, non-volatile memory of the BIOS. *Id.*

Notably, when evaluating the term “agent,” courts have found that the term does not connote sufficient structure to a person of ordinary skill. In *Synchronoss Techs., Inc. v. Dropbox Inc.*, No. 16-cv-00119, 2017 WL 6059302, at *10 (N.D. Cal. Dec. 7, 2017), the court found that the term “synchronization agent” invoked 35 U.S.C. § 112 ¶ 6, despite the fact that the term included the prefix “synchronization” because a person of ordinary skill in the art would only understand the term “agent” to be a “generic descriptor[] for software or hardware that . . . manages something.” Similarly, in *Joao Control & Monitoring Systems LLC v. Protect America, Inc.*, the Court evaluated the terms “intelligent agent,” “software agent,” and “mobile agent,” and ultimately concluded: “there was no single agreed-upon definition of the ‘agent’ terms which a skilled artisan would have understood—with reasonable certainty—based on the terms’ use in the claim language

alone.” No. 1-14-cv-134, 2015 WL 4937464, at *7 (W.D. Tex. Aug. 18, 2015).³

Ancora argues that an “agent” is simply a “software program or routine,” and thus not governed by 35 U.S.C. § 112 ¶ 6. However, as this Court recently recognized, the fact that certain terms may connote some type of software (or hardware running software) is not enough to remove them from the ambit of 35 U.S.C. § 112 ¶ 6. *Digital Retail Apps Inc. v. H-E-B, LP*, No. 6-19-cv-00167-ADA, 2020 WL 376664, at *5 (W.D. Tex. Jan. 23, 2020) (“[E]ven though ‘module’ refers to software, and ‘communication module’ certainly refers to software related to communication, this is nowhere near the ‘sufficiently definite structure’ required by Federal Circuit precedent.”).⁴

The remainder of the limitation at issue here fails to provide any further structure for the term “agent.” Rather, the limitation merely recites the function of the “agent,” i.e., to “set up a verification structure in the erasable, non-volatile memory of the BIOS.” While the claim describes what is being stored, i.e., the verification structure, and where it is being stored, i.e., the non-volatile memory of the BIOS, it does not recite any particular structure for doing so, nor does it describe how the “agent” fits in structurally with the other components of the system.

2. The specification and the prosecution history confirm that “agent” does not connote structure.

The scope of the functions performed by the “agent” in claims 1 and 18 vary significantly.

³ The term “agent” has only been found to connote sufficient structure in certain instances where the claims and specification explained how the “agent” interacted with other components in a way that informed its structural character. *See, e.g., Genband USA v. Metaswitch Networks*, No. 2:14-cv-33-RG-RSP, 2015 WL 4722185, at *17-18 (E.D. Tex. Aug. 7, 2015).

⁴ Other courts have found terms like “software” not to indicate sufficiently definite structure, and are therefore subject to 112 ¶ 6. *See, e.g., Cypress Lake Software, Inc. v. Samsung Elecs. Am., Inc.*, 382 F. Supp. 3d 586, 614-17 (E.D. Tex. 2019) (finding “code for” terms subject to 112 ¶ 6); *Global Equity Mgmt. (SA) Pty. Ltd. v. Expedia, Inc.*, No. 2:16-cv-00095-RWSRSP, 2016 WL 7416132, at *29-*30 (E.D. Tex. Dec. 22, 2016) (finding “program code for configuring . . .” subject to 112 ¶ 6); *Verint Sys. Inc. v. Red Box Recorders Ltd.*, 166 F. Supp. 3d 364, 379-381 (S.D.N.Y. 2016) (finding “first computer application operative” subject to 112 ¶ 6).

While claim 1 requires that the “agent” perform the function of “sett[ing] up a verification structure in the erasable, non-volatile memory of the BIOS,” claim 18 requires that the agent not only set up the verification structure by “encrypting license information using the pseudo-unique key” and “storing the encrypting license information,” but also that the agent must (1) “extract[] license information from [a] software program,” and (2) “verify[] the application software program.”⁵ By ascribing so many different functions to the term “agent,” the ’941 Patent, as in *Williamson*, clearly uses the term “agent” as a black box to describe generic software (or software running on hardware) for performing numerous tasks. The specification fails to provide guidance because the term was added during prosecution and does not otherwise appear outside of the claims.

Claim 1 as originally drafted did not include the term “agent,” and the Examiner rejected the claim because it did not disclose the necessary hardware to “add, remove and modify a license record” in the BIOS Memory. Ex. 5 (June 22, 2001, Final Rejection, pp. 3-6). In response, the applicants amended claim 1 to include the “using an agent” limitation. Ex. 6 (Response to Final Office Action, Nov. 14, 2001, p. 2). The applicants later noted that “the present invention proceeds against conventional wisdom The BIOS area is not considered a storage area for computer applications.” Ex. 8 (Response, Feb. 5, 2002, p. 6). The Examiner allowed the claims because the applicants had apparently addressed the problem by proposing a novel solution that purportedly did not exist in prior art, i.e., an “agent” that could set up the verification structure in BIOS:

[T]he closest prior art systems, singly or collectively, do not teach licensed programs running at the OS level interacting with a program verification structure stored in the BIOS to verify the program using the verification structure Further, it is well known to those of ordinary skill of the art that a computer BIOS is not setup to manage a software license verification structure. ***The present invention overcomes this difficulty by using an agent to set up a verification***

⁵ The Examiner noted that the “agent” did not perform the function of “acting on the application software program,” Ex. 9 (Notice of Allowance dated Mar. 28, 2002, Reasons for Allowance, pp. 2-3); however, claim 18, as written, suggests that the “agent” performs this function as well.

structure in the erasable, non-volatile memory of the BIOS.

Ex. 9 (Notice of Allowance dated Mar. 28, 2002, Reasons for Allowance, p. 4) (emphasis added).

The fact that the term “agent” does not connote sufficiently definite structure in the context of the claims is consistent with the applicants’ position and the Examiner’s understanding that the agent performs a function that is allegedly novel and did not exist in prior art. *See Techno View IP, Inc. v. Facebook Techs., LLC*, No. 17-386-CFC-CJB, 2018 WL 6427874, *4 (D. Del. Dec. 7, 2018) (noting that means-plus-function claiming applies to method claims where “the structural limitation clearly constitutes the point of novelty in the invention.”).

3. The associated algorithm is disclosed at 6:18-28 in the patent.

Because the term “agent” does not connote sufficiently definite structure, “the court must determine what structure, if any, disclosed in the specification corresponds to the claimed function” of setting up the verification structure. *Williamson*, 792 F.3d at 1351. “Structure disclosed in the specification qualifies as ‘corresponding structure’ if the intrinsic evidence clearly links or associates that structure to the function recited in the claim.” *Id.* at 1352. “Computer-implemented means-plus-function claims are indefinite unless the specification discloses an algorithm to perform the function associated with the limitation.” *Digital Retail*, 2020 WL 376664, at *3. The only portion of the specification that resembles an algorithm states:

Setting up (18) the verification structure includes the steps of: establishing or certifying the existence of a pseudo-unique key in the first non-volatile memory area; and establishing at least one license-record location in the first or the second nonvolatile memory area. Establishing a license-record includes the steps of: forming a license-record by encrypting of the contents used to form a license-record with other predetermined data contents, using the key; and establishing the encrypted license-record in one of the at least one established license-record locations (e.g. 10-12 in FIG. 1).

’941 Patent at 6:18-28 (emphasis added). To the extent Ancora argues that 6:18-28 is not the corresponding structure, then the claims must be rendered indefinite, as there is nothing else in the

specification that resembles the required algorithm.

B. “set up a verification structure”

Defendants’ Proposal	Plaintiff’s Proposal
“forming a structure by encrypting a license record using a pseudo-unique key for each computer”	plain and ordinary meaning

The parties dispute whether the limitation “set up the verification structure” requires encrypting a license record using a pseudo-unique key for each computer. In describing the invention in the specification and distinguishing the present invention from the prior art, the applicants made clear that this is indeed a requirement of the claims.

The “Summary of the Invention” states that “[t]he *present invention* relates to a method of restricting software operation within a license limitation,” and that “[t]his method *strongly relies* on the use *of a key and of a record.*” Ex. 1 (’941 Patent) at 1:38-42 (emphasis added). The Summary further states that the verification structure “is implemented by *encrypting the license record* (or portion thereof) *using said key* (or portion thereof) exclusively or in conjunction with other identification information) [*sic*] as an encryption key.” *Id.* at 1:62-65 (emphasis added).⁶

The remainder of the specification is consistent. The specification states that setting up the verification structure involves encrypting a license record using a pseudo-unique key:

Setting up (18) the verification structure includes the steps of: *establishing or certifying the existence of a pseudo-unique key* . . . and establishing at least one license-record location Establishing a license-record includes the steps of: forming a license-record by *encrypting of the contents used to form a license-record . . . using the key.*

Ex. 1 (’941 Patent) at 6:18-28 (emphasis added).

During prosecution, the applicants distinguished the present invention from the prior art

⁶ “Statements that describe the invention as a whole, rather than statements that describe only preferred embodiments, are more likely to support a limiting definition of a claim term.” *C.R. Bard, Inc. v. U.S. Surgical Corp.*, 388 F.3d 858, 864 (Fed. Cir. 2004).

based on the features that the Defendants have incorporated in their proposed construction. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1317 (Fed. Cir. 2005) (“[T]he prosecution history can often inform the meaning of the claim language by demonstrating how the inventor understood the invention and whether the inventor limited the invention in the course of prosecution, making the claim scope narrower than it would otherwise be.”). *Poly-Am., L.P. v. API Indus., Inc.*, 839 F.3d 1131, 1136 (Fed. Cir. 2016) (“While disavowal must be clear and unequivocal, it need not be explicit. For example, an inventor may disavow claims lacking a particular feature when the specification describes ‘the present invention’ as having that feature.”) (internal citations omitted).

The pending claims were initially rejected as being anticipated by Ginter. The Examiner noted that Ginter, *inter alia*, disclosed selecting a program, setting up a verification structure, and verifying the program using the verification structure. Ex. 3 (Office Action, Dec. 20, 2000, pp. 2-3). In their response, the applicants argued that Ginter did “not anticipate the present invention” because it “d[id] not disclose, among other things, setting up a verification structure and verifying the program using the verification structure.” Ex. 4 (Response, May 23, 2001, p. 6).

Ginter discloses distributing software using either “Stationary Objects” or “Traveling Objects.” *See id.* at 6-7. In the “Stationary Objects” method, the software is encrypted prior to distribution using a key that is unique to the target computer. *See id.* The applicants argued that the “Stationary Object” mechanism was deficient, because encrypting the software by a key unique to the target computer hindered a business model where the distributor did not know the final target computer. *Id.* at 7. In the “Traveling Object” method, the same key is used to encrypt the software for any computer and the key is incorporated in the distributed software. The applicants argued that this method was also deficient. *Id.* at 7-8. Using the same key repeatedly and incorporating it in software compromised security.

The applicants further argued that their claimed invention overcomes deficiencies associated with Ginter because their method encrypted the verification record at the user's computer using a key that was unique to the computer. *Id.* at 8-9. The applicants explicitly distinguished both methods identified in Ginter, arguing that the “***present invention*** differs from and overcomes the deficiencies associated with the stationary object and traveling object methods described in Ginter.” *Id.* at 8. The applicants stressed that by encrypting the verification record using a key unique to the computer, it did not encounter the problems faced by Ginter's solution:

In the ***present invention***, a unique key is stored in the first non-volatile memory of the computer. A software program in the volatile memory of the computer is selected. ***A license record is extracted from the software program and encrypted using the unique key stored in the computer*** (see new independent claim 20). Thus, the software program is not machine bound as is required by the stationery object method, nor is the same key used over and over to encrypt the software as is the case with the traveling object. ***In the present method, the verification structure is formed by using a unique key for each computer and license record information in the software.***

Id. (emphasis added). The applicants continued:

In contrast to Ginter et al., ***the present invention*** provides a system and method which not only enables free distribution of the software (such as happens in retail stores, and software companies that ship millions of copies), that overcomes the problems with the stationary object in Ginter et al., but also does not suffer from the limitations of incorporating the key in the distributed data as is the case with the traveling object of Ginter et al.

Id. at 9 (emphasis added).

Following another rejection, the applicants amended the claims, stating: “Specifically, claim 1 has been amended to recite that the verification structure is stored in an erasable, non-volatile memory area of BIOS.” Ex. 6 (Response to Final Office Action, Nov. 16, 2001, p. 7). The amended claims were then rejected as obvious based on Misra and Ewertz. Ex. 7 (Office Action, Jan. 15, 2002, p. 4). In their Response, the applicants conceded that Misra was similar to their invention in significant aspects. Notably, despite having removed the “unique key” language from

the preamble of claim 1, the applicants continued to acknowledge that it was a requirement of the invention and present in the pending independent claims: “Thus, the client system ID of Misra, is similar to the pseudo-unique key recited in claims 1 and 20.” Ex. 8 (Response, Feb. 5, 2002, p. 3) (emphasis in original). The applicants further conceded that the “identification information of Ewertz corresponds to the pseudo-unique key stored in the first non-erasable, non-volatile memory as recited in claims 1 and 20.” *Id.* at 4 (emphasis in original). In light of the patentee’s explanation, encrypting the verification structure using a pseudo-unique key (which is a superset of a unique key) of the computer is an integral aspect of the invention.

“The public is entitled to take the patentee at his word” on the scope of his invention. *See Honeywell Int’l, Inc. v. ITT Indus., Inc.*, 452 F.3d 1312, 1318 (Fed. Cir. 2006); *Bell Atlantic Network Servs., Inc. v. Covad Commc’ns Group, Inc.*, 262 F.3d 1258, 1273 (Fed. Cir. 2001) (finding that the term “channel” must be separated by frequency based on statement in prosecution that “in the present invention, the transmission bandwidth of channels . . . are controlled for various modes”); *Fenner Investments, Ltd. v. Celco P’ship*, 778 F.3d 1320, 1325 (Fed. Cir. 2015) (“[T]he interested public has the right to rely on the inventor’s statements made during prosecution. . . .”). Here, when attempting to overcome prior art during prosecution, the applicants explicitly stated that their invention requires forming a structure by encrypting a license record using a pseudo-unique key for each computer. Accordingly, Defendants’ construction should be adopted.

C. “memory of the BIOS”

Defendants’ Proposal	Plaintiff’s Proposal
“a memory that: (i) stores the BIOS; (ii) is not recognized by an operating system as a storage device; and (iii) does not have a file system”	“non-volatile memory” = memory whose data is maintained when the power is removed “of the BIOS” = plain and ordinary meaning

According to the patent, one advantage of the purported invention is that it relies upon the “memory of the BIOS,” which is more difficult to access and thus less susceptible to tampering.

Ex. 1 ('941 Patent) at 1:46-2:5; 3:4-14; 4:45-48. Consistent with this concept, the applicants clarified during prosecution that, unlike other memories, BIOS memory was not recognized by an operating system as a storage device and did not have a file system. In an effort to overcome the prior art, the applicants purposefully and unequivocally restricted the scope of the claims to exclude memory that is recognized as an OS as a storage device and has a file system. Defendants' proposal reflects how the applicants circumscribed their purported invention. Ancora's proposal, in contrast, ignores the statements the applicants made to the USPTO and equates "memory of BIOS" to any non-volatile memory irrespective of the fact that such memory may be easily accessible.

1. The applicants disclaimed a "memory of the BIOS" that is recognized by an operating system as a storage device or that contains a file system.

The applicants made several statements during prosecution to distinguish the BIOS memory from other types of computer memory. These statements were crucial to the Examiner's allowance. *See Kaken Pharm. Co. v. Iancu*, No. 2018-2232, 2020 WL 1222728, at *6 (Fed. Cir. Mar. 13, 2020) (stating that a patentee "is bound by its arguments made to convince the examiner that claims . . . are patentable"). As noted above, the Examiner rejected the pending claims as obvious based in part on the combination of Misra and Ewertz. The Examiner noted that Misra disclosed "storing a license record in non-volatile memory." Ex. 7 (Office Action, Jan. 15, 2002, p. 5). Misra taught storing license records in a local cache memory and not BIOS memory. Ex. 2 (Misra, 3:22-24). The Examiner turned to Ewertz for its disclosure of "expanding BIOS memory to store identification and/or configuration data such as software licenses," Ex. 7 (Office Action, Jan. 15, 2002, p. 6), noting that "it would have been obvious to one of ordinary skill to use the BIOS to store licenses in the Misra et al. system as they teach of users [*sic*] storing license data in persistent- non-volatile storage ('146, column 12, lines 8-27)." *Id.*

To overcome this rejection, the applicants disputed the Examiner’s contention that the Misra and Ewertz combination rendered the claims obvious. They argued that “Misra fails to teach using the BIOS of a computer to store the license ID” and that using BIOS memory to store such data would violate “conventional wisdom”:

Moreover, the *present invention* proceeds against conventional wisdom in the art. Using BIOS to store application data such as that stored in Misra’s local cache for licenses is not obvious. The BIOS area is not considered a storage area for computer applications.

Ex. 8 (Response, Feb. 5, 2002, p. 6) (emphasis added). The applicants further stated: “An ordinary skilled artisan would not consider the BIOS as a storage medium to preserve application data for at least two reasons.” *Id.* First, the applicants explained that “an ordinary person skilled in the art would not consider the BIOS for any operation, including writing to the BIOS” because the operating system (OS) did not view BIOS memory as a storage device:

First, *OS does not support this functionality* and is not recognized as a hardware device like other peripherals *There is no OS support whatsoever to write data to the system BIOS. Therefore, an ordinary person skilled in the art would not consider the BIOS as a possible storage medium. Furthermore, it is common that all peripheral devices in the PC are listed and recognized by the OS except for the BIOS.* This supports the fact that the BIOS is not considered a peripheral device.

Id. (emphasis added). The applicants distinguished memory recognized and used with an OS from the BIOS memory. According to the applicants, because the OS would not provide support for writing data to the BIOS memory, a person of ordinary skill would not consider writing to a BIOS memory, such as taught in Ewertz, when implementing a system based on the teaching of Misra.

Second, the applicants argued that a skilled artisan would not consider the BIOS as a storage medium because there is no file system associated with the BIOS:

Second, *no file system is associated with the BIOS.* Every writable device connected to the PC is associated with an OS file system to arrange and manage data structures *No such file system is associated with the BIOS.* This is further evidence that OS level application programmers would not consider the BIOS as a storage medium for license data.

Id. (emphasis added). A file system organizes and controls how data is stored or retrieved. Without it, a computer does not have a way to determine where one piece of data ends or the next begins. Thus, a memory with a file system is easier to access than one without a file system.

The applicants' explicit representations about the lack of OS support and a file system for BIOS mirror the benefits of using BIOS memory articulated in the specification:

An important advantage in utilizing non-volatile memory such as that residing in the BIOS is that the required level of system programming expertise that is necessary to intercept or modify commands, interacting with the BIOS, is substantially higher than those needed for tampering with data residing in volatile memory such as hard disk.

Ex. 1 ('941 Patent) at 3:4-9. The specification continues, "Furthermore, there is a much higher cost to the programmer, if his tampering is unsuccessful, i.e. *if data residing in the BIOS* (which is necessary for the computer's operability) is inadvertently changed by the hacker. This is too high of a risk for the ordinary software hacker to pay." *Id.* at 3:9-14 (emphasis added).

The applicants set bounds on the scope of the claims by relying on the essential features of their claimed invention during prosecution. These representations—including that the BIOS is not recognized as a storage device and has no file system—were key to the applicants' argument that using the BIOS as a local storage area for licenses is "against conventional wisdom in the art." Ex. 8 (Response, Feb. 5, 2002, p. 6). In the Notice of Allowance, the Examiner noted that the OS level programs were not interacting with structures stored in BIOS memory. Ex. 9 (Notice of Allowance dated Mar. 28, 2002, Reasons for Allowance, p. 4). Even the Federal Circuit found that "those representations [were] made in distinguishing prior art." *Ancora Techs.*, 744 F.3d at 736. When allowing the claims, the Examiner clearly understood the claims to be limited to BIOS memory that is not recognized by OS as a storage device and does not have a file system.

When faced with a rejection, the applicants argued that a skilled artisan would not consider BIOS memory to store the verification structure because the OS did not view such memory as a

storage device and because such memory lacked a file system. Ancora cannot now to broaden the claims to recapture memory systems that were distinguished. *See, e.g., ViaSat, Inc. v. Space Sys./Loral, Inc.*, No. 3:12-CV-00260-H, 2013 WL 3927729, at *20 (S.D. Cal. May 29, 2013) (agreeing that the prosecution history requires that the term “cache” exclude “DRAM” where the applicants argued that “[a] cache memory is a portion of memory made of high-speed static RAM (SRAM) instead of the slower and cheaper dynamic RAM (DRAM) used for main memory.”); *Arendi S.A.R.L. v. Google LLC*, 882 F.3d 1132, 1136 (Fed. Cir. 2018) (finding a disclaimer in the prosecution history where “the applicant amended the claims and explained what was changed and why, and the examiner confirmed the reasons why the amended claims were deemed allowable”).

2. The “memory of the BIOS” is the memory that stores the BIOS.

The claim term “memory of BIOS” itself implies that it stores the BIOS. The “Summary of the Invention” points to “a conventional computer having a conventional BIOS module” that a person of ordinary skill would understand to store the BIOS. Ex. 1 (’941 Patent) at 1:46-47. The patent mentions that the “encrypted license record is stored in another (second) non-volatile section of the BIOS,” thereby implying that the first portion stores BIOS. *Id.* at 1:65-67. In seeking to broaden the term to capture any memory, Ancora reads out the language “of the BIOS.”

D. “verifying the program using at least the verification structure”

Defendants’ Proposal	Plaintiff’s Proposal
“confirming through an operating system (OS) level application whether a program is licensed using at least the verification structure”	“confirming whether a program is licensed using at least the verification structure”

While the parties agree that the construction requires “confirming whether a program is licensed using at least the verification structure,” Ancora improperly ignores that during prosecution, the applicants depicted the “present invention” as an operating system (OS) level program that could not operate at the BIOS-level. *See Kaken*, 2020 WL 1222728, at *6 (stating that

a patentee “is bound by its arguments made to convince the examiner that claims . . . are patentable”).

When attempting to overcome a rejection based on Misra and Ewertz, the applicants argued that Misra operated at the OS-level, whereas Ewertz operated at the BIOS-level. The applicants contended that the two levels were mutually exclusive and that, like Misra, their invention also operated at the OS-level:

Furthermore, there is no suggestion or motivation to combine Misra and Ewertz in the manner suggested by the Office Action. BIOS is a configuration utility. ***Software license management applications, such as the one of the present invention, are operating system (OS) level programs.*** Therefore, BIOS programs and software licensing management applications do not ordinarily interact or communicate because when the BIOS is running, the computer is in a configuration mode, hence OS is not running. Thus, ***BIOS and OS level programs are normally mutually exclusive.***

Ex. 8 (Response, Feb. 5, 2002, p. 5) (emphasis added). The applicants reiterated that “Ewertz teaches that writing to the BIOS area is performed by BIOS routines” and that “Misra teaches a licensing system that is OS level based” and the two are incompatible: “Thus, the systems described by Misra and Ewertz are an OS program and a BIOS program, respectively, that cannot run at the same time.” *Id.*

These statements were critical to the allowance of the claims. When allowing the claims, the Examiner relied upon the applicants’ characterization that their license verification application (i.e., the present invention) runs at the OS-level and not at the BIOS-level:

[T]he key distinction between the present invention and the closest prior art, is that the Misra et al., and Ginter et al. systems and the Ewertz et al. system run at the operating system and BIOS level, respectively. More specifically, the closest prior art systems, singly or collectively, ***do not teach licensed programs running at the OS level*** interacting with a program verification structure stored in the BIOS to verify the program using the verification structure and have a user act on the program according to the verification.

Ex. 9 (Notice of Allowance dated Mar. 28, 2002, Reasons for Allowance, p. 4) (emphasis added).⁷

With these clear statements in prosecution that the license management application is an OS-level program, the applicants disclaimed non-OS-level applications. *Poly-Am*, 839 F.3d at 1136 (“While disavowal must be clear and unequivocal, it need not be explicit. For example, an inventor may disavow claims lacking a particular feature when the specification describes ‘the present invention’ as having that feature.”) (internal citations omitted); *see also Honeywell*, 452 F.3d at 1318 (“The public is entitled to take the applicants at his word.”). Ancora cannot now gain the benefit of claim scope that the applicants disclaimed.

E. “acting on the program according to the verification”

Defendants’ Proposal	Plaintiff’s Proposal
<p>“(i) allowing the use of the program if licensed or (ii) restricting the program’s operation if not licensed, using an operating system (OS) level application”</p> <p>The antecedent basis for “the verification” is the earlier step of “verifying the program using at least the verification structure from the erasable nonvolatile memory of the BIOS”</p>	<p>plain and ordinary meaning</p>

The specification clearly explains what the “acting on the program according to the verification” limitation means in the context of claim 1. After discussing the verifying step—during which the licensed software program’s license record is compared with the license record in the BIOS—the specification states:

Acting (20) on the program includes the step of: restricting the program’s operation with predetermined limitations if the comparing yields non-unity or insufficiency “Restricting the program’s operation with predetermined limitations” may include actions such as erasing the software in volatile memory, warning the license

⁷ The specification confirms that the invention uses an OS-level application to perform the verification. The “Summary of the Invention” describes that a “license verifier application” performs the claimed invention. Ex.1, ’941 Patent at 2:15-19. The fact that the applicants employed the term “application” implies that they believed that their invention would be run at the OS-level.

applicant/user, placing a fine on the applicant/user through the billing service charges collected at the license bureau (if applicable), or scrambling sections of the BIOS of the computer (or of functions interacting therewith).

Ex. 1 ('941 Patent) at 6:40-52. Defendants' construction is entirely consistent with this explanation: if the program is licensed, the use of the program is allowed; if it is not, the use of the program is restricted. The preamble of the claim, which recites "[a] method of restricting software operation within a license," offers further support for this construction. *See id.* at 6:58-59. As noted in Defendants' arguments regarding the limitation "verifying the program" limitation, the applicants' statements that the invention is an OS-level application are binding. Thus, the application that performs the "acting" step must also be an OS-level application.

Finally, as explained in Defendants' arguments below regarding the order of the steps of claim 1, the antecedent basis for "the verification" of "acting on the program according to the verification" is the earlier step of "verifying the program using at least the verification structure from the erasable nonvolatile memory of the BIOS." Indeed, Ancora acknowledged as much in the HTC Litigation. *See* Ex. 15 (HTC Litigation, ECF No. 59) at 4.

F. "license"/"license record"

Defendants' Proposal	Plaintiff's Proposal
The entire preamble of the claim is limiting.	Only "including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area" in the preamble is limiting.
"license" means "a legal contract between a software provider and a user that specifies the rights of the user to use, distribute or resell the software"	"license" = preamble is non-limiting; if held to be limiting, "license" means "authorization or verification to run"
"license record" means "a record associated with a program with information for verifying that the program is licensed"	"license record" = "a record associated with a licensed program with information for verifying that licensed program"

1. The preamble of the claim is limiting.

As discussed above with respect to the limitation "verifying the program using at least the

verification structure,” the parties agree that claim 1 requires confirming whether a program is licensed. Indeed, Ancora’s construction requires “*confirming whether a program is licensed* using at least the verification structure.” Accordingly, even if the preamble of claim 1 is not found to be limiting, the term “license” needs to be construed, as it is a limitation of the claim. However, here, the entirety of the preamble of claim 1 is limiting.

A preamble limits the scope of the claims if it is “necessary to give life, meaning, and vitality” to the claims. *Pitney Bowes, Inc. v. Hewlett-Packard Co.*, 182 F.3d 1298, 1305 (Fed. Cir. 1999); *see also C.W. Zumbiel Co. v. Kappos*, 702 F.3d 1371, 1385 (Fed. Cir. 2012) (the “preamble constitutes a limitation when the claim(s) depend on it for antecedent basis, or when it is ‘essential to understand limitations or terms in the claim body’”).

The preamble of claim 1 recites:

A method of *restricting software operation within a license for use with a computer* including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area; the method comprising the steps of:

Ancora concedes that the second half of the preamble (portion underlined above) is limiting but disputes that the first half (portion italicized above) limits the claim. Because the entire preamble is necessary to understand the claim, the entire preamble should be found to be limiting. The term “license” is recited in the preamble and gives life and meaning to the claim. To begin with, “license” gives meaning to the phrase “license record,” as well as other terms derived from “license.”⁸ Consistent with Defendants’ understanding that “license” defines the claims, the Federal Circuit characterized the claims as “methods for verifying that a software program on a

⁸ Claim 1 requires a “license record” to be stored and used to verify a program and invokes terms recited in the preamble. Further, many of the dependent claims include derivatives of the software license in the preamble: claim 2 recites a “license authentication bureau,” claim 3 recites a “request-for-license” as well as “license record”; claim 6 recites “licensed-software-program” and “license record”; and claims 7, 8 and 9 recite a “license record.”

computer is not there without authorization, but is *licensed* to be there.” 744 F.3d at 733 (emphasis added). Further, the preamble states that the “license” is “for use with a computer” and that the BIOS is part “*of the computer*.” Thus, both the license and the BIOS are part of the *computer*, and both of these preamble terms are necessary to give life to the claims.

2. License

Although the term “license” is not defined in the specification or the claims, it is a common word that is easily understood. *Phillips*, 415 F.3d at 1314 (“In some cases, the ordinary meaning of claim language as understood by a person of skill in the art may be readily apparent even to lay judges, and claim construction in such cases involves little more than the application of the widely accepted meaning of commonly understood words.”). Specifically, “license” carries its normal meaning in the software context. That meaning is reflected in numerous technical and lay dictionaries. For example, the Microsoft Press Computer Dictionary defines a “license agreement” as “[a] legal contract between a software provider and a user specifying the rights of the user regarding the software.” Ex. 16 at DEFS_ANCORA00002101. Similarly, Webster’s New College Dictionary defines “license” as “[o]fficial or legal permission to do or own a specified thing.” Ex. 17 at DEFS_ANCORA00002177. Ancora’s extrinsic evidence confirms Defendants’ construction. Ancora relies on the same Microsoft Dictionary as Defendants for other terms. The American Heritage Dictionary that Ancora relies upon defines license as “official or legal permission.” Ex. 18 at ANCORA_00003375. When providing a tutorial on the ’941 Patent in the Apple Litigation, the inventor, Mr. Mullor, stressed the legal aspect of the term “license”:

The last item we have here is what we call a license, software license. And the concept – *our concept of software license is very similar to the concept of a license plate*. The idea we had is, just like a car, you may have it, but you are not allowed to operate it without a license plate.

We can have the same concept with computers and software. If you don't have the special license plate in the computer, you can't run the software associated with that license plate.

Ex. 13, (Apple Litigation, ECF 104) at 11:18-12:2 (emphasis added).

Significantly, if the claims are not limited to a license based on the construction proposed by the Defendants, their scope would be unbounded and could be met by *any verification* mechanism. Indeed, that is Ancora's aim. Under Ancora's proposal, if a program is downloaded from a server on a computer and checked to ensure that its contents were not corrupted during transmission, then these steps could potentially meet the "verifying that licensed program" part of Ancora's construction. The downloaded program could then be run on the computer. Under this scenario, there would be *no verification* that the program was actually *licensed* to run on the computer. Ancora's proposal would eviscerate the key concept of checking for "license for use with a computer" from claim 1 and, in effect, would make the patented method less secure in direct contradiction with the supposed purpose of the invention. There is no reason to deviate from the technical dictionary definitions that mirror how a person of ordinary skill would interpret the term in the context of the '941 Patent.

3. License Record

As illustrated above, the parties dispute whether the record has information "for verifying that a program is licensed" or can simply contain information "for verifying a licensed program." This is no mere issue of semantics, because verifying "a licensed program" presumes the program is already licensed, and the program itself is then verified. Merely confirming that a program was not corrupted during transmission could potentially meet Ancora's construction. However, the purpose of the alleged invention is to utilize a "license record" to verify that a program is, in fact, licensed to run on a computer, as reflected in Defendants' construction.

G. Order of Steps

Defendants' Proposal	Plaintiff's Proposal
The “verifying the program” step and “acting on the program” step of claim 1 must occur, in order, after the “selecting a program” step and “using an agent” step	The steps do not need to be performed in a specific order.

Method steps must be performed in the order recited where such order is dictated by the plain language of the claim. *See Mantech Envtl. Corp. v. Hudson Envtl. Servs., Inc.*, 152 F.3d 1368, 1375-76 (Fed. Cir. 1998) (“the sequential nature of the claim steps is apparent from the plain meaning of the claim language and nothing in the written description suggests otherwise”). Courts may look to the claim language to determine if, as a matter of logic or grammar, the steps must be performed in the order written. *See Altiris, Inc. v. Symantec Corp.*, 318 F.3d 1363, 1369-70 (Fed. Cir. 2003).

The logic and grammar of the claim 1 dictate that Step 1 of claim 1, i.e., the “verifying the program” step, and Step 4 of claim 1, i.e., the “acting on the program” step, must occur, in order, after Steps 1 and 2 of “selecting a program” and “using an agent.” Specifically, Step 3 requires “verifying *the program* using at least *the verification structure from the erasable non-volatile memory of the BIOS.*” For this verification to occur requires (1) knowing what selected program is being verified (which occurs in Step 1), and (2) having access to a verification structure already stored in the erasable non-volatile memory (which occurs in Step 2). Thus, Step 3 must necessarily follow the completion of the first two steps.

Step 4 requires “acting on the program *according to the verification.*” Because the verification occurs in Step 3, Step 4 necessarily follows Step 3. In the HTC Litigation, even Ancora agreed that “[t]he antecedent basis for ‘the verification’ [in Step 4] is the earlier step of ‘verifying the program using at least the verification structure from the erasable non-volatile memory of the BIOS’ [of Step 3].” Ex. 15 (HTC Litigation, ECF No. 59) at 4.

In the Apple Litigation, the court determined that no order of steps is required. Ex. 14 (Apple Litigation, ECF No. 107) at 19-20. However, the court’s analysis focused only on Steps 1 and 2. *Id.* Defendants agree with the Apple court that Steps 1 and 2 of the claim can be performed in any order. However, the grammar and logic of claim 1 dictates that Step 3 and Step 4 must be performed, in order, *after* Steps 1 and 2 (which can be performed in any order).

H. “BIOS”

Defendants’ Proposal	Plaintiff’s Proposal
An acronym for <u>B</u> asic <u>I</u> nput / <u>O</u> utput <u>S</u> ystem. It is the set of essential startup operations stored in ROM that run automatically when a computer is turned on, which test hardware, starts the operating system, and support the transfer of data among hardware devices	An acronym for Basic Input/ Output System. It is the set of essential startup operations that run when a computer is turned on, which test hardware, starts the operating system, and support the transfer of data among hardware devices

Although the Apple court construed the term BIOS, the construction omitted two key concepts. Defendants respectfully ask this Court to supplement that construction. *See Biedermann Motech GmbH v. Acme Spine, LLC*, No. CV 06-3619 SJO, 2007 WL 6210841, at *8 (C.D. Cal. Aug. 31, 2007) (explaining that it may be that a prior construction “needs to be tweaked, remaining consistent with the original definition, to resolve a dispute in another case.”). The Parties agree regarding the majority of the construction except whether the BIOS is “stored in ROM” and “run automatically”—two issues not evaluated by the Apple court. The intrinsic and extrinsic record both confirm that these are integral aspect of BIOS.

1. “BIOS” is stored in ROM.

The specification explicitly and repeatedly discloses that “BIOS” is “stored in ROM.” The invention is described throughout as comprising “a conventional computer having a conventional BIOS module in which a key was embedded *at the ROM section thereof*, during manufacture.”

Ex. 1 ('941 Patent) at 1:45–47 (emphasis added).⁹ Likewise, the technical dictionaries from the relevant time frame consistently describe BIOS as being stored in ROM.¹⁰ Accordingly, “BIOS” should be construed to be stored in ROM.

2. “BIOS” runs automatically when a computer is powered on.

The “BIOS” is also intended to run automatically—without user or software intervention—when the computer is powered on, because, the “BIOS” is “necessary for the computer’s operability.” Ex. 1 ('941 Patent) at 3:11–12. In other words, the “BIOS” automatically runs at the exact time the computer turns on—before the computer processes any other software or input/output connections. Ancora’s expert, Ian Jestice, explained “[a]t the time the computer is first started, BIOS automatically performs the initial steps necessary to boot the operating system.” Ex. 19 (Apple Litigation, ECF 95-5) at ¶ 10. Without the “BIOS,” the computer “has no inherent knowledge of what it is connected to for input or output, or where to find the first instructions to execute.” *Id.* at ¶ 11. “Before the advent of BIOS, these parameters had to be set manually by the user,” however, “[f]or decades . . . BIOS has been implemented to perform these steps automatically.” *Id.* at ¶ 12. In fact, Mr. Jestice confirmed that he was “not aware of any computer available for retail sale . . . in 1998 when the ‘941 application was filed that require[d] manual initialization . . . because they all include BIOS which performs the initialization steps

⁹ See Ex. 1, '941 Patent at 2:17–18 (“from the ROM section of the BIOS”), 2:31–32 (“stored in the ROM portion of the BIOS”), 2:38 (“stored in the ROM portion of the BIOS”), 4:51 (“a ROM section of the BIOS”), 4:61 (“a ROM section of the BIOS”); 5:13–14 (“the ROM section of the BIOS”).

¹⁰ See Ex. 16 (Microsoft Press Computer Dictionary (3rd ed. 1997)) (“stored in read-only memory (ROM)”); Ex. 20 (Que’s Computer & Internet Dictionary (6th ed. 1995)) (“encoded in read-only memory (ROM)”); Ex. 21 (Dictionary of Computer and Internet Terms (5th ed. 1996)) (“stored on a ROM chip”); Ex. 22 (Webster’s New World Dictionary of Computer Terms (6th ed. 1997)) (“encoded in read-only memory (ROM)”); Ex. 23 (Free On-Line Dictionary of Computing) (“stored in ROM”); Ex. 24 (Dictionary of Computing (4th ed. 1996)) (“held in read-only memory (ROM)”).

automatically at start-up.” *Id.* at ¶ 13. Consistent with Ancora’s own expert’s understanding, “BIOS” runs automatically when a computer is powered on.

I. “selecting a program residing in the volatile memory”

Defendants’ Proposal	Plaintiff’s Proposal
“running a program in the volatile memory”	plain and ordinary meaning

In the Microsoft Litigation, Ancora proposed that the step of “selecting a program residing in the volatile memory” should be construed as “running a program in the volatile memory,” noting that “[t]his construction is most consistent with the context of the ’941 Patent disclosure.” Ex. 10 (Microsoft Litigation, ECF No. 101) at 16-17. Defendants agree with Ancora’s prior construction, which, as Ancora previously noted, is supported by the intrinsic record. *See* Ex. 1 (’941 Patent) at 1:61-62 (“the specified program is licensed to run on the specified computer.”). Ancora now tries to avoid its prior construction by asserting “plain and ordinary meaning,” with no explanation as to what that “plain and ordinary meaning” is or how it differs from Ancora’s previous construction. The intrinsic evidence has not changed, only Ancora’s position has. Ancora should not be allowed to shield itself from its prior construction.

J. “program”

Defendants’ Proposal	Plaintiff’s Proposal
“a set of instructions that can be executed by a computer”	“a set of instructions for a computer”

Here again Defendants have set forth a position previously advocated for by Ancora, and supported by the intrinsic and extrinsic evidence, and once again Ancora has shifted its position. As Ancora argued in both the Apple and Microsoft Litigations, “[e]very person of skill in the computer field knows exactly what a ‘program’ is: ‘a set of instructions that can be executed by a computer,’” i.e., Defendants’ construction. Ex. 10 (Microsoft Litigation, ECF No. 101) at 14; Ex. 11 (Apple Litigation, ECF No. 94) at 12. Ancora previously acknowledged that this definition is consistent with the technical dictionaries at the time of the alleged invention. *Id.* (quoting the

Microsoft Computer Dictionary as defining “program” as “a sequence of instructions that can be executed by a computer.”). *Id.*

Although “every person of skill in the computer field knows exactly what a program is,” Ancora now presents a different construction. Ancora’s current construction appears to be based on a single statement made by the Federal Circuit in the Apple Litigation:

And the district court explained that, although the term “program” may have many different meanings depending on the context, “to a computer programmer” a program is merely a “set of instructions” for a computer. That clear meaning governs here, we conclude, because there is nothing sufficient to displace it.

744 F.3d at 734-35 (citation omitted). But Ancora misinterprets the Federal Circuit’s opinion. The Federal Circuit was not evaluating whether or not a program “can be executed”—the point of contention between the parties here—but rather, whether, as the District Court had found, a program is limited to an “application.” *Id.* at 737 (“We conclude that the district court erred in construing ‘program’ to mean ‘a set of instructions for software *applications* that can be executed by a computer.’”). The “clear meaning” the Federal Circuit found governing was, as the quotation marks in the opinion suggest, a “set of instructions.” While the Federal Circuit found that the term “applications” should not have been included by the district court, it did not take issue with the remainder of the construction, which had been previously advocated for by Ancora.

K. “volatile memory”

Defendants’ Proposal	Plaintiff’s Proposal
“memory whose data is not maintained when the power is removed”	“memory whose data is not maintained or becomes inaccessible when the power is removed”

The Federal Circuit unambiguously set out the definition of “volatile memory”: “Most importantly, there is no dispute that the term[] ‘volatile memory’ [has] a meaning that is clear, settled, and objective in content. *Both parties and the district court agreed* that, as a general matter, ‘[t]o one of ordinary skill in the art, *a volatile memory is memory whose data is not*

maintained when the power is removed.” 744 F.3d at 737 (emphasis added). Ancora also adopted the *same* construction in the HTC Litigation, and an analogous construction in the Microsoft Litigation (“memory that is not maintained when the power is removed from the storage system.”). Ex. 10 (Microsoft Litigation, ECF No. 101) at 14; Ex. 12 (Apple Litigation, ECF No. 100-1) at 1; Ex. 15, (HTC Litigation, ECF No. 59) at 24. There is no reason to deviate from that construction. Notably, Ancora’s construction for “volatile memory” encompasses memory whose data becomes inaccessible when the power is removed. This construction would subsume “non-volatile memory” as well, which also becomes inaccessible when the power is removed, further demonstrating that Ancora’s construction cannot be correct.

L. “first non-volatile memory area of the computer”

Defendants’ Proposal¹¹	Plaintiff’s Proposal
“read only memory area of the computer”	plain and ordinary meaning

The specification describes what the “first non-volatile memory area of the computer” means in the context of claim 7. Notably, the parties are in agreement that “non-volatile memory” means “memory whose data is maintained when the power is removed.” The ’941 Patent discloses both a first non-volatile memory and a second non-volatile memory. *Id.* at 2:1-5. The specification distinguishes the first and second non-volatile memories by indicating that the second non-volatile memory is “*unlike* the first non-volatile memory section” in that “the data in the *second* non-volatile memory may optionally be erased or modified (using E2PROM manipulation commands), so as to enable to add, modify or remove licenses.” *Id.* (emphasis added). Thus, the first non-volatile memory area must have no option to be erased or modified, i.e., it must be read only memory (ROM). *See id.* at 4:50-51; 5:12-13.

¹¹ Defendants Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc. do not join Defendants LG Electronics Inc. and LG Electronics U.S.A., Inc. in the construction of this term.

Dated: March 20, 2020

Respectfully submitted,

/s/ Collin W. Park

Winstol D. Carter, Jr.
Texas Bar No. 03932950
winn.carter@morganlewis.com
Thomas R. Davis
Texas Bar No. 24055384
thomas.davis@morganlewis.com
Elizabeth M. Chiaviello
Texas Bar No. 24088913
elizabeth.chiaviello@morganlewis.com
MORGAN, LEWIS & BOCKIUS LLP
1000 Louisiana Street, Suite 4000
Houston, Texas 77002-5006
T. 713.890.5000
F. 713.890.5001

Collin W. Park (admitted *pro hac vice*)
collin.park@morganlewis.com
District of Columbia Bar No. 470486
MORGAN, LEWIS & BOCKIUS LLP
1111 Pennsylvania Avenue, NW
Washington, D.C. 20004-2541
T. 202.739.3000
F. 202.739.3001

**Attorneys for Defendants LG Electronics
Inc. and LG Electronics U.S.A., Inc.**

/s/ Melissa R. Smith

Melissa R. Smith
Texas State Bar No. 24001351
melissa@gillamsmithlaw.com
GILLAM & SMITH, LLP
303 South Washington Avenue
Marshall, Texas 75670
Phone: (903) 934-8450
Fax: (903) 934-9257

Robert T. Haslam (rhaslam@cov.com)
Anupam Sharma (asharma@cov.com)
COVINGTON & BURLING LLP
3000 El Camino Real

5 Palo Alto Square, 10th Floor
Palo Alto, CA 94306-2112
Phone: (650) 632-4700
Fax: (650) 632-4800

Richard L Rainey (rrainey@cov.com)
Jared Frisch (jfrisch@cov.com)
COVINGTON & BURLING LLP
One CityCenter
850 Tenth Street, NW
Washington, DC 20001-4956
Phone: (202) 662.6000
Fax: (202) 662-6291

**Attorneys for Defendants Samsung
Electronics Co., Ltd., and Samsung
Electronics America, Inc.**

CERTIFICATE OF SERVICE

The undersigned hereby certifies that all counsel of record who are deemed to have consented to electronic service are being served with a copy of the foregoing document via the Court's CM/ECF system this March 20, 2020.

/s/ *Melissa R. Smith*
Melissa R. Smith